

基于滤波波束搜索算法的置换流水车间调度问题*

董海¹, 王瀚鹏²

(1. 沈阳大学 应用技术学院; 2. 沈阳大学 机械工程学院, 沈阳 110044)

摘要:【目的】研究解决置换流水车间中作业完成时间与常见到期日之间的绝对偏差最小化和中间库存最小化问题。【方法】首先,构建基于最小化绝对偏差和中间库存的混合整数线性规划模型,采用两阶段法对模型进行求解;其次,提出基于精英选择策略的滤波波束搜索算法(Filter beam search algorithm,FBSA),将精英选择策略与 FBSA 融合,通过对有“希望”的迭代节点进行局部搜索和全局搜索,用以确定筛选出的最优解。【结果】通过实例仿真对基于精英选择策略的 FBSA 与其他智能优化算法在最优解偏差百分比、平均错误率、改进百分比等 3 个指标进行对比,结果显示该算法在求解所提问题时具有高效性和稳定性。【结论】基于精英搜索策略的 FBSA 对处理置换流水车间调度问题有着良好的效果和优势。

关键词:滤波波束搜索算法;置换流水车间调度;混合整数线性规划模型;精英选择策略;两阶段法

中图分类号:O221;TP273

文献标志码:A

文章编号:1672-6693(2022)01-0041-11

置换流水车间调度问题(Permutation flow shop scheduling problem,PFSP)是实际生产过程中的一类典型调度问题,被广泛应用于自动化、工业生产等领域中解决工件排序问题。为提高企业竞争力,生产者在选取调度方案时,需考虑的因素并不唯一的,往往要求方案能够在生产过程中使多个目标同时达到最优。

NEH 启发式算法是处理 m 台机器和 n 个工件流水车间调度问题的最常用方法^[1]。有限等待置换流水车间调度研究始于 2006 年,Fondreville 等人^[2-3]证明了最小化 Makespan 问题是强 NP-难的,他们针对等待时间上下限相同且最小化最大拖期问题给出了多项式可解的特例,并提出了分支定界法求解该问题。Dhouib 等人^[4]建立了最小化拖期工件数和顺序依赖的安装时间数学模型,采用模拟退火算法(Parthasarathy and Rajendran's simulated annealing algorithm,PRSA)求解模型。Sharma 等人^[5]试图通过减少序列来降低 m -机器流水车间环境的时间复杂度,建立了一个基于最优的或近似最优的最大完工时间数学模型,采用 CDS 算法求解该模型。Santucci 等人^[6]利用有限生成群的概念,定义了一般组合空间的抽象代数微分变异算子,该算子可通过原始随机泡泡排序算法对空间进行重新排列;并提出了求解置换问题的离散差分进化算法(Differential evolution algorithm,DE),将它应用于具有总流时间准则的 PFSP。张先超等人^[7]针对 PFSP,改进了基本量子进化算法,设计了一种变参数量子进化算法。Lu 等人^[8]提出了一种基于最大适应度函数的混合粒子群算法,通过新的动态搜索策略来增强离散粒子群的局部搜索能力,通过仿真分析证明了提出的算法能有效地解决双资源约束的多目标调度问题。Lin^[9]提出了一种有效的基于离散生物地理学的混合优化算法求解 PFSP,使最大完工时间最小化。Rossi 等人^[10]针对具有总流时间最小化准则的 PFSP,提出了一种新的简单的构造性启发式方法解决该问题。Shao 等人^[11]提出一种多邻域 IG 算法求解最小化最大完成时间。Zheng 等人^[12]针对模糊分布式混合流水车间调度问题(Distributed hybrid flow shop scheduling problem)提出一种分布估计算法与 IG 相结合的协同进化算法以优化总延迟时间和鲁棒性。

近些年来,采用 NEH 算法和优先规则的元启发式算法用来求解 PFSP 成为新的研究方向。如改进布谷鸟搜索算法(Improved Cuckoo search algorithm)^[13]、混合猴群搜索算法(Hybrid monkey search algorithm)^[14]、DE^[15]等。Ying 等人^[16]提出一种带有双向移位定时算法(Multi-start simulated annealing with bi-directional shift timetabling algorithm)的多启动 PRSA 以解决最大完工时间问题。张洪亮等人^[17]针对带有学习效应的多

* 收稿日期:2021-06-18 修回日期:2021-09-09 网络出版时间:2022-02-09 17:44

资助项目:国家自然科学基金(No. 71672117);中央引导地方科技发展资金计划(No. 2021JH6/10500149)

第一作者简介:董海,男,教授,博士,研究方向为网络制造化,E-mail:478318377@qq.com

网络出版地址:https://kns.cnki.net/kcms/detail/50.1165.n.20220209.1212.016.html

阶段混合流水车间以解决最小化最大完工时间和总能耗问题。秦旋等人^[18]提出基于共生生物搜索算法(Symbiotic organisms search algorithm)与局部搜索策略结合的混合共生生物搜索算法以求解 PFSP。

由上述介绍可知,目前针对流水车间调度问题大多是采用新型智能算法并结合局部搜索策略来提升算法的效率和质量。滤波波束搜索算法(Filtered beam search algorithm,FBSA)是一种基于波束搜索技术改进的启发式智能优化算法,具有搜索能力强、搜索时间快、参数少等特点。滤波波束搜索是波束搜索的扩展,而波束搜索技术在路径规划^[19]、通信、无人机^[20]等领域都取得了良好的应用效果,但是鲜有学者将它用于流水车间调度优化领域。为此,本文针对 PFSP 中作业完成时间与常见到期日之间的绝对偏差最小化、作业等待时间最小化和最小化中间库存成本等问题,建立了混合整数线性规划(Mixed integer linear programming,MILP)模型,并提出了一种适应环境特定特征的 FBSA 用于解决该问题。

1 PFSP 描述及高维多目标模型的建立

1.1 PFSP 描述

PFSP 是许多实际流水线生产调度问题的简化模型,是典型的 NP-难问题。问题描述为: n 个独立的工件依次依据相同的路线被 m 台机器进行加工,每台机器上工件的加工顺序相同,每个工件同一时间只能在 1 台机器上加工,每台机器同一时间只能加工 1 个工件,每个工件在每台机器上的加工时间已知,期望获得工件的最优加工序列使得多个性能指标共同达到最优。本文主要研究作业最大延迟/提前时间和中间库存两目标最小化的 PFSP。

1.2 混合整数线性规划模型的建立

PFSP 可以使用 p_{ik} 表示作业 i 在机器 k 上的处理时间, d 表示所有工作的截止日期, C_{ik} ($i=1,\dots,n;k=1,\dots,m$)表示机器 k 上作业 i 的完成时间, E_i 和 T_i ($i=1,\dots,n$)分别表示作业 i 的提前和延误时间, z_{ij} ($i=1,\dots,n,j=i+1,\dots,n$)表示作业 i 在作业 j 之前($z_{ij}=1$)或之后($z_{ij}=0$)。本文基于 PFSP 建立如下模型及约束:

$$\min \sum_{i=1}^n (E_i + T_i); \quad (1)$$

$$\text{s. t. } T_i \geq C_{im} - d, i=1, \dots, n; \quad (2)$$

$$E_i \geq d - C_{i1}, i=1, \dots, n; \quad (3)$$

$$C_{i1} \geq p_{i1}, i=1, \dots, n; \quad (4)$$

$$C_{i,k+1} \geq p_{i,k+1} + C_{ik}, i=1, \dots, n, k=1, \dots, m-1; \quad (5)$$

$$C_{ik} \geq p_{ik} + C_{jk} - Mz_{ij}, i=1, \dots, n, j=i+1, \dots, n, k=1, \dots, m; \quad (6)$$

$$C_{jk} \geq p_{jk} + C_{ik} - M(1 - z_{ij}), i=1, \dots, n, j=i+1, \dots, n, k=1, \dots, m; \quad (7)$$

$$z_{ij} \in \{0, 1\}, i=1, \dots, n, j=i+1, \dots, n; \quad (8)$$

$$T_i, E_i \geq 0, i=1, \dots, n; \quad (9)$$

$$\sum_{i=1}^n \sum_{k=1}^{m-1} \omega_{ik} (C_{i,k+1} - p_{i,k+1} - C_{ik}); \quad (10)$$

$$\sum_{i=1}^n (C_{im} - C_{i1}); \quad (11)$$

$$\frac{\sum_{i=1}^n (C_{im} - C_{i1})}{n(d + p)} < 1. \quad (12)$$

其中:(1)式为提前和延误之和目标函数;(2),(3),(8),(9)式与 $T_i \geq \max\{C_{im} - d, 0\}$ 和 $E_i \geq \max\{d - C_{i1}, 0\}$ 对应,因此它们分别对每个工件的提前和延迟性施加了下限;(4)式表示第 1 台器上的作业完成时间必须大于或等于该机器上的作业处理时间;(5)式表示在两台连续机器上的作业完成时间之间,必须有足够的时间在两台机器中的第 2 台上处理作业;(6),(7)式确保在任何给定时间在每台机器上仅处理 1 个工件,其中 M 是足够大的正常数;(10),(11)式表示连续机器之间的作业等待时间的加权总和;(12)式表示假设以上问题的最佳值都为整数,连续机器之间的作业等待时间可转化为(12)式。

下式为最终目标函数:

$$n(d+p) \sum_{i=1}^n (E_i + T_i) + \sum_{i=1}^n (C_{im} - C_{i1}). \quad (13)$$

本文用两阶段法求解多目标问题。在(13)式中第 1 阶段该序列最大程度地减少了提前性和延误性,在第 2 阶段中同时保留了相同的序列及早期性和延误性的总和,从而使等待时间最小化。因此,两阶段方法包括解决两个 MILP 问题:第 1 个是找到最佳的早熟和拖延时间;第 2 个是限制提前和延迟的最佳时间以最大程度地减少等待时间。

2 FBSA

2.1 FBSA 描述

波束搜索是分支定界方法的一种改编,通过它可以逐级探索有希望的节点,在每个级别上探索的节点数称为波束宽度。在波束搜索技术中,每个级别的节点评估过程都是一个关键问题。通常使用评估函数来确定在哪个节点上继续搜索,需要在快速但较差的评估与对计算要求更高但更好的评估之间寻求良好的平衡。为了经济、快速地找到上述良好的平衡,引入滤波波束搜索。经滤波后的波束搜索过程在滤波阶段和波束选择阶段分别使用粗略和准确的评估。由父节点生成的所有节点均通过计算进行过滤(通常称之为本地评估功能),在第 2 阶段保留总宽度评估功能(通常称之为全局评估功能),对过滤器宽度进行过滤,以进行进一步的准确评估,其中最有前途的节点将添加到部分解决方案中。在平行波束宽度路径上重复此过程,直到达到完整的时间表。因此,节省的解决方案数量等于波束宽度的大小,最终将具有最佳目标函数值的解决方案将作为最终计划。

2.2 编码方式

采用基于升序排列(Ranked order value, ROV)规则的随机编码方式^[21]。对于 n 个工件 m 台机器的 PFSP, 将每个染色体对应转换为 n 个工件的一个排列。例如,有一条染色体 $X_i = \{x_{i1}, x_{i2}, \dots, x_{im}\}$, 则它的对应工件初始排列为 $1, \dots, l, \dots, n$, 其中 l 表示工件号($0 < l < n+1$)。每个 l 出现次数为 1, 使染色体每个位置分量对应为工件初始排列中的一个工件号;再将染色体的位置分量从小到大进行排序,将最小的分量位置赋予 ROV 值 1, 其次将第 2 小的分量位置赋予 ROV 值 2, 依次类推,直到所有分量位置都赋予一个唯一的 ROV 值;最后,将染色体位置分量对应初始排列的工件号,按照 ROV 值进行重新排列,即得到一个工件排列,在解码时可以按照工件排列顺序转化为一个调度方案。

2.3 精英选择策略

利用精英选择策略,结合列表调度算法理论,将每次迭代适应函数值较好的个体选择作为首先调度的个体,从而使精英个体可以优先完成调度。具体步骤如下。

步骤 1:通过以下绩效函数对各工件在所有机器上的完成时间进行评估:

$$\Gamma_j = \frac{1}{2} \sum_{k=1}^m p_{jk} + \frac{1}{2} \sum_{k=1}^m |C_{\sigma_{i-1}, k+1} - C_{\sigma_{i-1}, k} - p_{jk}|; \quad (14)$$

步骤 2:将计算得出的适应函数值个体最好的工件添加入序列,再将未添加序列的工件通过(14)式计算,再次添加适应度最好的工件,以此类推,直至所有的工件都添加至序列;

步骤 3:重新计算该序列,最大程度减少每台机器之间的作业等待时间;

步骤 4:确定最后 1 台机器上作业的完成时间,最大程度地减少作业的早期性和延误性;

步骤 5:通过计算所有其他机器中作业的完成时间,重新建立调度的紧凑性。

3 基于精英选择策略的 FBSA

3.1 改进算法描述

本文提出一种基于精英选择策略的 FBSA,具有先进的优先级评估功能。该算法与 B&B 算法相似,此计算过程将构造一个搜索树,其中每个节点由部分序列组成,子节点是通过在父节点的末尾添加未完成计划的作业之一而获得的。仅保留最有前途的 γ 节点(即波束宽度)用于下一次迭代。

本文提出的基于精英选择策略的 FBSA 由 n 个不同级别的几个节点组成。其中: S_l^k 表示第 k 次迭代的第 l 个节点的部分序列, $l \in [1, \gamma]$ 。每个节点由 k 个作业的部分序列 $S_l^k, S_l^k := (S_{1,l}^k, \dots, S_{k,l}^k)$ 和 $n-k$ 个未计划作业的集合 $U_l^k (U_l^k := \{u_{1,l}^k, \dots, u_{n-k,l}^k\})$ 组成。对于每次迭代,通过在序列末尾添加集合 U_l^k 中的一个作业,从每个部

分序列 S_l^k 中创建 $n-k$ 个子节点, 选择最佳的 γ 子节点作为下一次迭代的节点的部分序列, 即 $S_l^{k+1}, \forall l \in \{1, \dots, \gamma\}$ 。这一算法的步骤为: 步骤 1, 初始化; 步骤 2, 迭代, 其中又分为 3 个子步骤: 分支、节点评估和节点选择; 步骤 3, 最终评估。为了阐明分支和候选选择阶段, 图 1 显示了一个简单示例, 其中包含 5 个作业且 $\gamma=2$ 。以下详细说明本文提出的基于精英选择策略的 FBSA 中的各个步骤。

3.1.1 初始化(步骤 1) 第 1 个节点由作业 j^* 组成, 该作业 j^* 在最后 1 台机器上的完成时间最少, 且加权的空闲时间为 ω_j , 将这些总和表示为 ξ_j , 即 $j^* := \arg \min_j \{\xi_j\}$; 如果是平局, 则选择最小的 ω_j 作业, 其中:

$$\xi_j := \sum_{i=1}^m p_{ij} + \omega_j = \sum_{i=1}^m p_{ij} + \frac{n-2}{4} \sum_{i=2}^m \left(\frac{m \sum_{i'=1}^{i-1} p_{i'j}}{i-1} \right), \forall j \in [1, n]。$$

3.1.2 迭代(步骤 2) 1) 分支。每个节点 S_l^k 被分支以形成 $n-k$ 个子节点, 通过在部分序列 S_l^k 的末尾添加 $u_{v,l}^k$, 从节点构造子节点 $v (v \in [1, n-k])$ 。迭代次数 k 中子节点的数目是 $\gamma(n-k)$ 。

2) 节点评估。如前所述, 子节点根据权重两个分量的索引进行评估, 其一为“遗传”代码: 该部分测量子节点的遗传后代, 以此能够比较具有不同排序作业(即来自不同节点)的子节点, 显然 S_l^k 节点的每个子节点具有相同的遗传密码; 其二为最后 1 个元素的影响, 即测量在部分序列末尾插入的最后 1 个作业的影响。

3) 节点选择。在迭代 k 的 $\gamma(n-k)$ 个子节点中, 最好的 γ 个被保留为迭代 $k+1$ 的节点集。即在迭代 k 中, 具有 G_{α}^k 指标最低值的 γ 节点 ($\forall v \in \{1, \dots, n-k\}, l \in \{1, \dots, \gamma\}$) 被选择用于下一次迭代。

3.1.3 最终评估(步骤 3) 在上一次迭代中选择节点的总提前和延迟性, 即评估节点 $S_l^{n-1} (\forall l \in \{1, \dots, \gamma\})$ 。产生最小的总提前或延迟的序列是 FBSA 的最终序列。

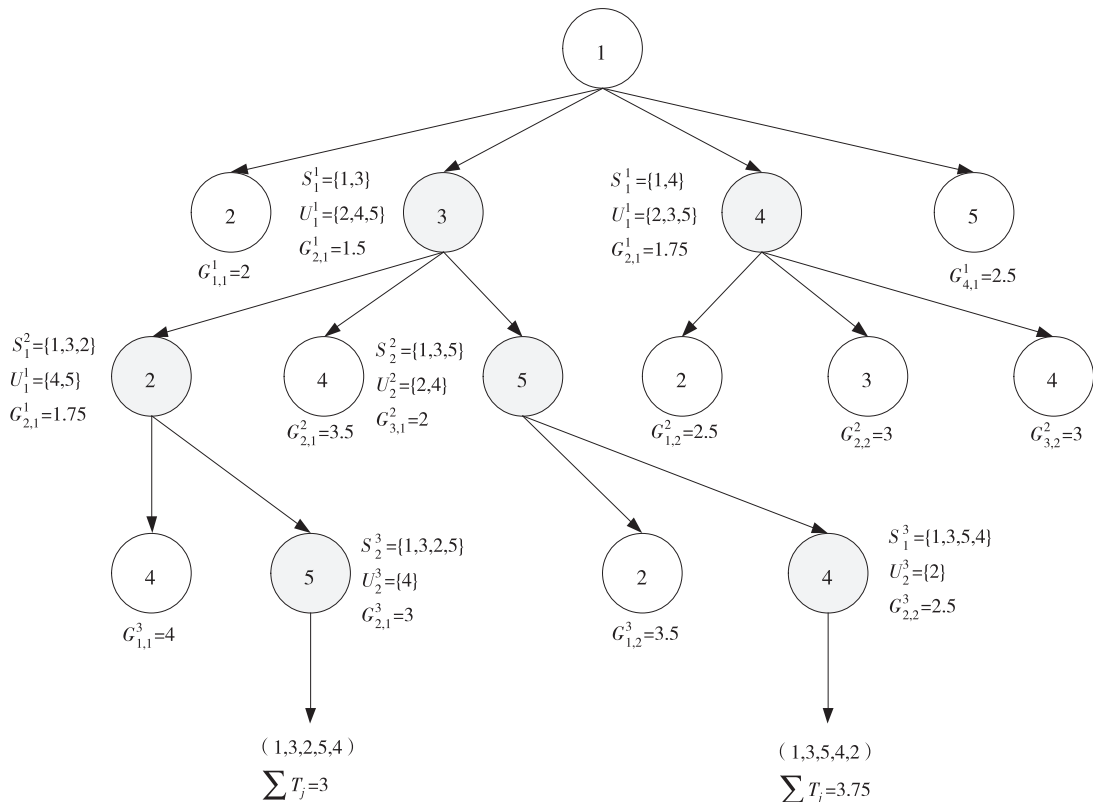


图 1 基于精英选择策略的 FBSA 示例

Fig. 1 The example of FBSA based on elite selection strategy

3.2 局部搜索策略

由于多目标作业车间解的空间较为复杂, 因此设计一种有效的局部搜索策略能够提高解的质量, 对算法的性能有着至关重要的作用。对首次迭代的候选解集进行局部搜索, 过滤次等解集, 剩余解集进行全局搜索, 提高算法搜索效率(表 1)。

以上为算法对首次迭代解集进行局部搜索、过滤次等分支,保留有希望的分支以确定各机器上第 1 件加工工件顺序,同时保证工件在各台机器加工无等待时间。其中有希望的波束开始继续分支,并继续对有希望的分支进行局部搜索(表 2)。总之,局部搜索能通过快速简便的方式过滤次等解集,以便全局搜索能对有希望的解集进行搜索,从而得出最优解。

表 1 局部搜索策略 1

Tab. 1 Local search strategy 1

行号	伪代码
1	$P_i \leftarrow \sum_{k=1}^m p_{ik}$, 其中 $i=1, \dots, n$
2	$L \leftarrow \emptyset$
3	$\{j_1, \dots, j_n\} = \{1, \dots, n\}$ 使得 $(P_{j_1}, j_1) < \dots < (P_{j_n}, j_n)$
4	$j \in \{j_1, \dots, j_n\}$
5	$\sigma = (1, 2, \dots, n)$ 和 (σ_1, σ_j) 进行交换
6	$C_{\sigma_1, m} \leftarrow \max\{d, P_{\sigma_1}\} - p_{\sigma_1, m}$, 使 $k \leftarrow m-1$ 到 1 并返回上一步, 且 $C_{\sigma_1, k} \leftarrow C_{\sigma_1, k+1} - p_{\sigma_1, k+1}$
7	$L \leftarrow L \cup (\sigma, C)$

表 2 局部搜索策略 2

Tab. 2 Local search strategy 2

行号	伪代码
1	$l \leftarrow 2$ 到 n
2	如果 $l \leq 4$ 则让 $\bar{\alpha} = n - l + 1$, 否则则使 $\bar{\alpha} \leftarrow \{\alpha, n - l + 1\}$
3	使每个 $(\sigma, C) \in L$
4	$C_{\sigma_1, 1} \leftarrow p_{\sigma_1, 1}$, 即 $k \leftarrow 2$ 到 m 且 $C_{\sigma_1, k} \leftarrow C_{\sigma_1, k-1} + p_{\sigma_1, k}$
5	$i \leftarrow 2$ 到 $l-1$
6	$C_{\sigma_1, 1} \leftarrow C_{\sigma_{i-1}, 1} + p_{\sigma_1, 1}$, 使得 $k \leftarrow 2$ 到 m , $C_{\sigma_i, k} \leftarrow \max\{C_{\sigma_i, k-1}, C_{\sigma_{i-1}, k}\} + p_{\sigma_i, k}$
7	$\bar{\alpha} < n - l + 1$ 则进行下一步
8	$j \leftarrow l$ 到 n 且 $\Gamma_{\sigma_j} \leftarrow \frac{1}{2} P_{\sigma_j} + \frac{1}{2} \sum_{k=1}^{m-1} C_{\sigma_{l-1}, k+1} - C_{\sigma_{l-1}, k} - p_{\sigma_j, k} $
9	$\{j_1, \dots, j_{n-l+1}\} = \{l, \dots, n\}$ 使得 $(\Gamma_{\sigma_{j_1}}, \sigma_{j_1}) < \dots < (\Gamma_{\sigma_{j_{n-l+1}}}, \sigma_{j_{n-l+1}})$
10	否则则使 $j_1 = l, j_2 = l+1, \dots, j_{n-l+1} = n$

注: l 表示算法迭代次数,下同

3.3 全局搜索策略

在局部搜索后,首先对有希望的分支进行全局搜索,在精英选择策略的基础上对分支上的解集序列计算作业的完成时间,以及最后 1 台机器中所有作业的完成时间;然后计算其他机器的完成时间,旨在获得紧凑的解集序列以及减少机器之间的等待时间;最后对完成最大搜索次数后得出最优解集,将解集里的所有解代入(13)式进行评估,保留最优解,淘汰其他解,算法结束(表 3)。

3.4 算法框架

为了提高最优解集的收敛性和分布性改进算法的性能,在基础 FB SA 框架下提出了一种基于精英选择策略的 FB SA,它的算法流程如图 2 所示。

4 仿真实验

为了验证本文提出的基于精英选择策略的 FB SA 的有效性,采用文献[22]中的多个不同规模的典型算例进行测试,并将仿真结果与邻域搜索算法(Neighborhood search algorithm, NS)、PRSA、改进模拟退火算法(Zegordi's

simulated annealing algorithm, ZSA)、可变贪婪算法(Framinan and Leisten's variable greedy algorithm, FLVG)以及快速蚁群算法(Fast ant colony algorithm, FACO)进行比较。实验所用计算机配置为 Inter core i7 2.80 GHz,内存为 8 GB,编程语言为 Matlab。

表 3 全局搜索策略

Tab. 3 Global search strategy

行数	伪代码
1	$\bar{\sigma}$ 对 σ 的值进行复制并和 $(\bar{\sigma}_i, \bar{\sigma}_j)$ 的值进行交换
2	$C_{\bar{\sigma}_i,1} \leftarrow C_{\bar{\sigma}_{i-1},1} + p_{\bar{\sigma}_i,1}, k \leftarrow 2$ 到 m 且 $C_{\bar{\sigma}_i,k} \leftarrow \max\{C_{\bar{\sigma}_{i-1},k-1}, C_{\sigma_{i-1},k}\} + p_{\bar{\sigma}_i,k}$
3	$i \leftarrow l+1$ 到 n
4	$t \leftarrow i$ 到 n 且 $\Gamma_{\bar{\sigma}_i} \leftarrow \frac{1}{2}P_{\bar{\sigma}_i} + \frac{1}{2} \sum_{k=1}^{m-1} C_{\bar{\sigma}_{i-1},k+1} - C_{\bar{\sigma}_{i-1},k} - p_{\bar{\sigma}_i,k} $
5	与 $(\bar{\sigma}_t, \bar{\sigma}_{t_1})$ 值进行交换, t_1 是其中对应最小的索引 $\{(\Gamma_{\bar{\sigma}_t}, \bar{\sigma}_t) \mid t=i, \dots, n\}$
6	$C_{\bar{\sigma}_i,1} \leftarrow C_{\bar{\sigma}_{i-1},1} + p_{\bar{\sigma}_i,1}$, 即 $k \leftarrow 2$ 到 m , $C_{\bar{\sigma}_i,k} \leftarrow \max\{C_{\bar{\sigma}_{i-1},k-1}, C_{\sigma_{i-1},k}\} + p_{\bar{\sigma}_i,k}$
7	$k \leftarrow m-1$ 到 1 并返回上一步
8	$C_{\bar{\sigma}_i,k} \leftarrow \min\{\bar{C}_{\bar{\sigma}_{i+1},k} - p_{\bar{\sigma}_{i+1},k}, \bar{C}_{\bar{\sigma}_i,k+1} - p_{\bar{\sigma}_i,k+1}\}$
9	通过设置 $E_j = \max\{0, d - \bar{C}_{\bar{\sigma}_j}\}$ 和 $T_j = \max\{0, \bar{C}_{\bar{\sigma}_j} - d\}$, 其中 $j=1, \dots, n$, 计算(13)式的目标函数值 $f(\bar{\sigma}, \bar{C})$
10	对 σ 和 C 进行赋值且 $(\sigma, C) \in L$ 是最小的元素 $f((\sigma, C), \sigma_n)$ 。

其中,工件数量 n 从 $\{8, 10, 12, 15, 20, 25, 30, 40, 50, 75, 100\}$ 选取时,机器数量 m 从 $\{5, 10, 20\}$ 中选取。每台机器的作业处理时间用 $1 \sim 100$ 内的整数表示,作业的到期日期使用整数 $T_{MS} \left(1 - r - \frac{R}{2}\right) MS$ 和 $T_{MS} \left(1 - r + \frac{R}{2}\right)$ 上均匀随机生成的。其中: T_{MS} 是使用 Taillard 提出的完成时间下限找到问题的预计完成时间, R 为到期日期范围, r 为拖延因子。测试了 3 个级别的到期日范围即 $R=0.2, 0.6, 1.0$ 和 5 个级别的到期日密度即 $r=0.0, 0.2, 0.4, 0.6, 0.8$, 由此一共会产生 15 组到期日参数;设置了一个时限停止实验过程,时限设置为 $\left(\frac{1}{5}n + \frac{1}{2}\right) \times \left(\frac{n}{20}\right)^2 \times \frac{m}{5}$ 。偏差百分比为评价最佳的早期性和延误性的重要指标,且 $P_{DEV} = \frac{Z_h - Z_0}{Z_0} \times 100\%$, 其中: Z_0 为最佳解决方案, Z_h 为总解决方案, P_{DEV} 为偏差百分比。

表 4 显示了每个算法得出的最优解偏差百分比以及每个算法针对要解决的问题所对应的的作业数量

(n)和机器数量(m)生成最佳解决方案的次数。结果表明,FBSA、FLVG 和 FACO 始终生成最佳或接近最佳的解决方案。这些算法生成的结果在总早期和延误范围之内,并且能生成更多的最佳解决方案。FBSA 的平均最优解偏差百分比最低,并针对 9 个算例中的生成了 8 个最佳解决方案。NS 和 PRSA 的平均最优解偏差百分比超过 1%,生成最优解的算例少于一半。ZSA 算法的平均最优解偏差百分比最高,生成的最优解的算例最少。

表 5~7 显示了当 m 为固定值时,每个算法的最优解偏差百分比和生成最佳解决方案的次数。结果表明,FBSA 在每个方案上均表现最佳,与其他算法相比,FBSA 的平均最优解偏差百分比最低,并且生成的最优解决方案次数最多。同时,在 24 个算例中,FBSA 有 20 个算例平均最优解偏差百分比低于 1%。PRSA 次优,其所有

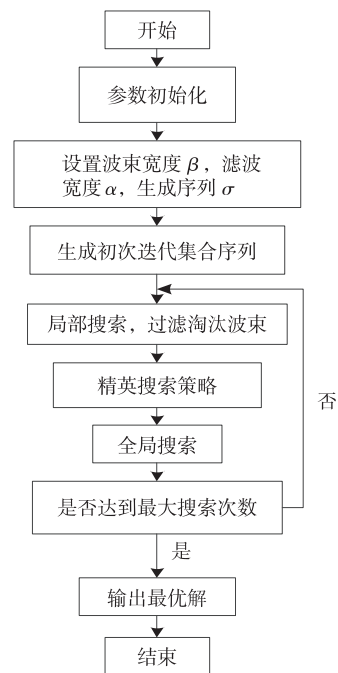


图 2 基于精英选择策略的 FBSA 流程图

Fig. 2 The flow chart of FBSA based on elite selection strategy

方案的平均最优解偏差百分比均低于 3.5%,每个方案的总平均最优解偏差百分比低于 2%。随着工作数量的增加,FLVG 和 FACO 算法的结果趋于恶化。ZSA 算法的结果很差,所有方案的平均最优解偏差百分比均高于 4%,并且没有生成最适合任何问题的解决方案。

表 4 最优解偏差百分比和最佳次数
Tab. 4 Deviation percentage and the best times of the optimal solution

作业数量(<i>n</i>)	机器数量(<i>m</i>)	NS	PRSA	ZSA	FBSA	FLVG	FACO
8	5	1.47%(106)	0.96%(115)	1.09%(109)	0.01%(149)	0.13%(144)	0.02%(146)
8	10	1.15%(112)	0.69%(123)	0.52%(117)	0.00%(150)	0.03%(148)	0.00%(148)
8	20	0.85%(109)	0.67%(123)	0.69%(103)	0.00%(149)	0.02%(147)	0.00%(151)
10	5	1.34%(86)	1.27%(87)	2.05%(56)	0.09%(138)	0.23%(126)	0.18%(126)
10	10	2.32%(65)	0.96%(104)	1.75%(67)	0.08%(144)	0.35%(128)	0.12%(133)
10	20	0.56%(83)	0.55%(108)	1.39%(58)	0.06%(141)	0.07%(136)	0.06%(135)
12	5	2.33%(57)	2.14%(74)	4.47%(27)	0.27%(115)	0.55%(97)	0.72%(82)
12	10	2.75%(48)	1.63%(75)	3.76%(26)	0.40%(108)	0.56%(95)	0.55%(83)
12	20	1.86%(42)	1.22%(87)	2.83%(29)	0.10%(133)	0.37%(101)	0.27%(97)

注:括号内数字为各算法生成最佳解决方案的次数,即最佳次数,下同

表 5 *m*=5 时最优解偏差百分比和最佳次数
Tab. 5 Deviation percentage and the best times of the optimal solution (*m*=5)

作业数量(<i>n</i>)	NS	PRSA	ZSA	FBSA	FLVG	FACO
15	2.87%(43)	3.35%(45)	6.84%(7)	0.55%(108)	1.62%(57)	1.44%(37)
20	3.94%(27)	3.27%(38)	9.15%(1)	0.65%(86)	3.27%(14)	3.04%(11)
25	2.67%(33)	3.26%(27)	9.37%(1)	0.56%(88)	4.25%(13)	3.53%(7)
30	3.13%(36)	2.95%(23)	10.42%(0)	0.85%(76)	4.37%(14)	4.24%(5)
40	2.53%(27)	2.66%(34)	11.16%(0)	0.91%(79)	7.04%(3)	4.57%(2)
50	3.65%(33)	2.85%(42)	11.54%(0)	0.86%(73)	6.67%(6)	5.23%(2)
75	5.03%(8)	2.27%(45)	11.15%(0)	0.79%(90)	5.96%(8)	4.65%(6)
100	6.63%(5)	1.65%(63)	11.66%(0)	1.05%(70)	4.77%(13)	4.72%(5)

表 6 *m*=10 时最优解偏差百分比和最佳次数
Tab. 6 Deviation percentage and the best times of the optimal solution (*m*=10)

作业数量(<i>n</i>)	NS	PRSA	ZSA	FBSA	FLVG	FACO
15	4.07%(26)	1.85%(67)	6.07%(8)	0.39%(103)	1.43%(63)	1.67%(33)
20	4.03%(15)	2.79%(42)	8.13%(3)	0.55%(95)	2.97%(26)	2.77%(17)
25	3.57%(23)	2.96%(35)	9.25%(3)	0.52%(94)	3.73%(23)	4.45%(5)
30	3.92%(12)	3.06%(34)	10.35%(0)	0.55%(101)	4.78%(8)	4.86%(6)
40	4.05%(21)	2.84%(33)	12.29%(0)	0.87%(88)	5.75%(7)	6.57%(2)
50	4.09%(15)	2.65%(33)	12.03%(0)	0.76%(89)	6.57%(9)	5.72%(3)
75	7.55%(4)	2.35%(42)	13.05%(0)	0.82%(91)	5.69%(5)	6.07%(5)
100	7.95%(3)	1.98%(43)	13.87%(0)	1.19%(85)	5.26%(8)	5.82%(5)

表 8 和表 9 显示了 R 和 r 对结果的影响。结果与先前的结果一致,对于每个不同 r 的方案,FBSA 的平均最优解偏差百分比均低于 2%,并且在所有算法中最低。除 PRSA 外,所有其他算法的平均最优解偏差百分比均大于或等于 4.36%。对于每个不同 R 的方案,FBSA 的平均最优解偏差百分比最低,并且每个值得平均值低于 1.16%,其他算法对任何 R 的平均最优解偏差百分比都高于 2%。

表 7 $m=20$ 时最优解偏差百分比和最佳次数Tab. 7 Deviation percentage and the best times of the optimal solution ($m=20$)

作业数量(n)	NS	PRSA	ZSA	FBSA	FLVG	FACO
15	2.42%(32)	1.68%(49)	4.37%(7)	0.22%(112)	0.87%(63)	1.05%(33)
20	2.95%(22)	2.14%(38)	6.43%(0)	0.40%(98)	1.95%(23)	2.55%(11)
25	3.03%(15)	2.74%(29)	7.63%(0)	0.35%(109)	3.07%(14)	3.73%(2)
30	3.87%(17)	3.07%(26)	9.74%(2)	0.53%(100)	3.96%(8)	4.61%(0)
40	4.09%(15)	3.29%(36)	9.97%(2)	0.73%(95)	4.32%(7)	5.73%(0)
50	3.95%(17)	3.24%(25)	11.85%(0)	0.59%(103)	6.14%(7)	6.67%(3)
75	4.99%(8)	2.27%(43)	11.84%(0)	1.04%(89)	5.57%(13)	6.25%(2)
100	7.05%(7)	2.28%(42)	12.45%(0)	1.07%(85)	4.64%(13)	5.67%(2)

表 8 $n=50, m=10$ 时基于 r 的最优解偏差百分比Tab. 8 Deviation percentage of the optimal solution based on r ($n=50, m=10$)

%

拖延因子(r)	NS	PRSA	ZSA	FBSA	FLVG	FACO
0.0	4.36	1.85	14.27	1.72	5.09	3.97
0.2	6.15	3.89	16.35	1.15	11.62	8.34
0.4	6.87	3.17	19.45	0.33	8.35	10.09
0.6	1.79	2.33	6.90	0.39	5.43	3.78
0.8	1.35	2.28	3.45	0.69	2.56	2.37

表 9 $n=50, m=10$ 时基于 R 的最优解偏差百分比Tab. 9 Deviation percentage of the optimal solution based on R ($n=50, m=10$)

%

到期日期范围(R)	NS	PRSA	ZSA	FBSA	FLGV	FACO
0.2	2.76	2.03	9.08	0.68	5.18	4.35
0.6	2.87	2.88	8.43	0.41	6.53	4.57
1.0	6.62	3.17	18.72	1.16	8.13	8.21

图 3 为各算法在 Taillard 基准测试的平均错误率(Average error rate, R_{AE})。 R_{AE} 的计算公式为:

$$R_{AE} = \frac{1}{k} \sum_{i=1}^k \frac{Z_i - Z_{opt}}{Z_i} \times 100\%。$$

其中: Z_i 为算法 i 求得平均完工时间, Z_{opt} 为目前已知的算例上界值, k 为算例的数量, FBSA 在 12 个算例中的 R_{AE} 为 1.35%, 远远低于其他算法的 R_{AE} , 表明了本文提出的基于精英选择策略的 FBSA 相对于其他 5 种算法而言具有更好的稳定性。

为了验证 FBSA 的性能, 研究了基于 FBSA 对其他 5 种算法的改进百分比(Improvement percentage, P_1)。

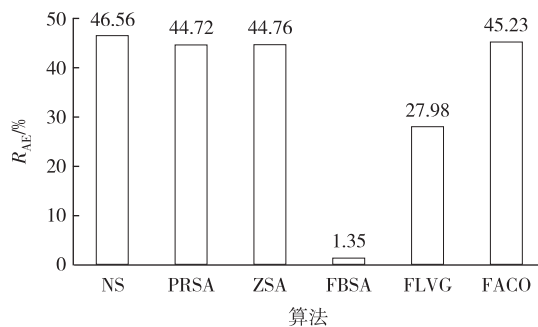


图 3 各算法在 Taillard 基准测试中的平均错误率

Fig. 3 Average error rate of each algorithm in Taillard benchmark test

P_1 的计算公式为:

$$P_1 = \frac{C_X - C_{\text{FBSA}}}{C_X} \times 100\%。$$

其中: C_X 为某一非 FBSA 的改进算法在不同规模算例下的平均完工时间, C_{FBSA} 为不同规模算例下 FBSA 的平均完工时间。由图 4 可以看出,与其他 5 种算法相比,本文提出的基于精英选择策略的 FBSA 的性能的到了很大提升。

在下面的实验中,将本文提出的基于精英搜索策略的 FBSA 与 PAPSO (Population adaptation particle swarm optimization)、HMM-FPA (Flower pollination algorithm based on the hormone modulation mechanism)、DWWO (Discrete water wave optimization)、TMIIG (Tabu-mechanism improved iterated greedy)、DPSOVND (Discrete particle swarm optimization)、IIGA (Improved iterated greedy algorithm) 等算法进行对比。进行 Friedman 检验以对算法进行排名,Friedman 的测试基于 SPSS 软件来实现。图 5 和图 6 均显示 FBSA 在运行 1 秒后在算法收敛性指标上要明显优于其他 6 种智能启发式算法。

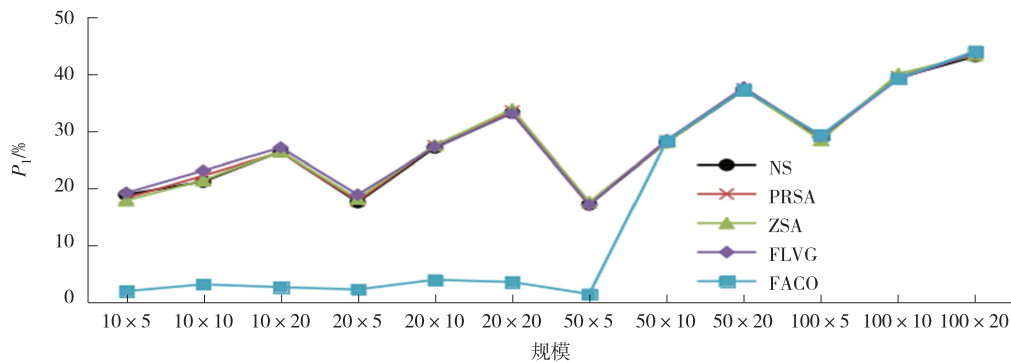
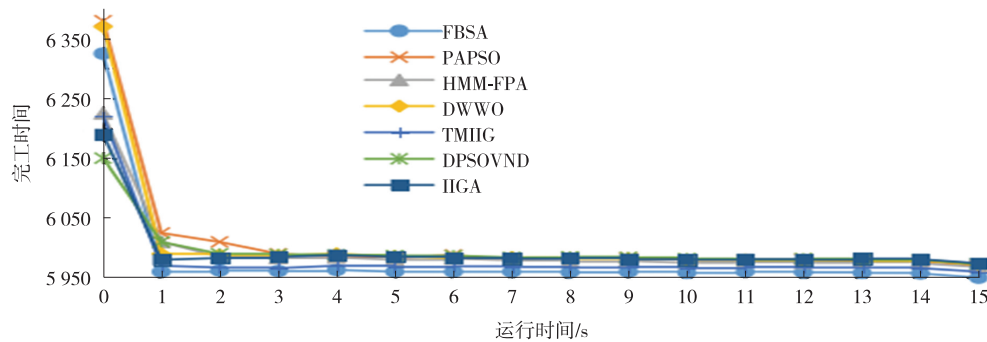


图 4 基于 FBSA 对其他算法的改进百分比

Fig. 4 Improvement percentage of other algorithms based on FBSA



注:完工时间指工作完成时间的相对长短,无具体时间单位,下同

图 5 TA060 算例收敛图

Fig. 5 TA060 convergence graph

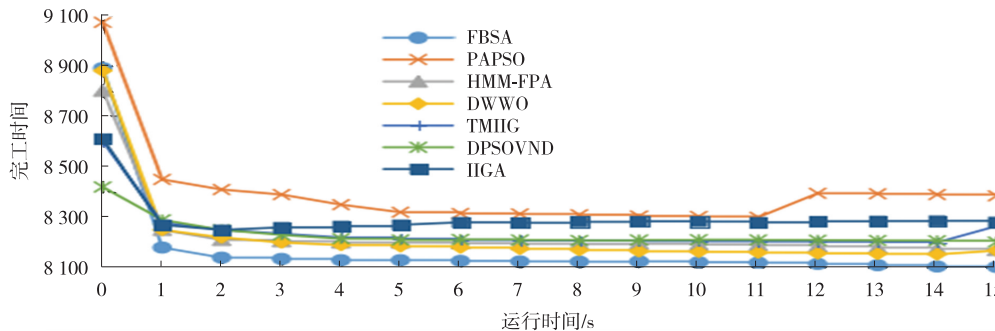


图 6 TA080 算例收敛图

Fig. 6 TA080 convergence graph

5 结束语

本文提出了一种基于精英搜索策略的 FBSA,用于最小化置换流水车间中作业完成时间与常见到期日期之间的绝对偏差,并在对以上目标优化的同时,最大程度减少车间的中间库存成本。为此,本文首先建立基于最小化绝对偏差和中间库存的 MILP 模型,并采用两阶段法,用它解决置换流水车间中作业完成时间与常见到期日期之间的绝对偏差最小化和中间库存最小化问题;然后,本文提出了基于精英选择策略的 FBSA,并引入局部搜索和全局搜索策略提高算法的搜索能力;最后,本文通过对置换流水车间测试实例、实际应用的仿真,并与多种智能优化算法进行比较验证,结果显示在最优解偏差百分比、平均错误率、改进百分比等方面,本文提出的算法均优于其他智能优化算法,且表明该算法在求解所提问题的高效性和稳定性。因此,本文提出的基于精英搜索策略的 FBSA 在解决 PFSP 上能够获得更好的解,且更适用于解决大规模问题,对处理 PFSP 有着良好的效果与优势。

参考文献:

- [1] PAN Q K, WANG L, GAO L. A chaotic harmony search algorithm for the flow shop scheduling problem with limited buffers [J]. *Applied Soft Computing Journal*, 2011, 11(8): 5270-5280.
- [2] FONDREVELLE J, OULAMARA A, PORTMANN M C. Permutation flowshop scheduling problems with maximal and minimal time lags [J]. *Computers & Operation Research*, 2006, 33(6): 1540-1556.
- [3] FONDREVELLE J, OULAMARA A, PORTMANN M C, et al. Permutation flow shops with exact time lags to minimise maximum lateness [J]. *International Journal of Production Research*, 2009, 47(23): 6759-6775.
- [4] DHOUB E, TEGHEM J, LOUKIL T. Lexicographic optimization of a permutation flow shop scheduling problem with time lag constraints [J]. *International Transactions in Operational Research*, 2013, 20(2): 213-232.
- [5] SHARMA M, SONI R, CHAUDHARY A, et al. Analyzing complexity using a proposed approximation algorithm MDA in permutation flow shop scheduling environment [C]//SATAPATHY S C, BHATT Y C, JOSHI A, et al. *Proceedings of the International Congress on Information and Communication Technology ICICT 2015, Volume 2*. Singapore: Springer, 2016: 121-130.
- [6] SANTUCCI V, BAIOLLETTI M, MILANI A. Algebraic differential evolution algorithm for the permutation flow shop scheduling problem with total flow time criterion [J]. *IEEE Transactions on Evolutionary Computation*, 2016, 20(5): 682-694.
- [7] 张先超, 周泓. 变参数量子化算法及其求解置换流水车间调度问题中的应用 [J]. *计算机集成制造系统*, 2016, 22(3): 774-781.
ZHANG X C, ZHOU H. Variable parameter quantum-inspired evolutionary algorithm and its application in permutation flow-shop scheduling problem [J]. *Computer Integrated Manufacturing System*, 2016, 22(3): 774-781.
- [8] LU C, GAO L, LI X Y, et al. Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm [J]. *Journal of Cleaner Production*, 2017, 144: 228-238.
- [9] LIN J. A hybrid discrete biogeography-based optimization for the permutation flow shop scheduling problem [J]. *International Journal of Production Research*, 2016, 54(16): 4805-4814.
- [10] ROSSI F L, NAGANO M S, SAGAWA J K. An effective constructive heuristic for permutation flow shop scheduling problem with total flow time criterion [J]. *The International Journal of Advanced Manufacturing Technology*, 2017, 90(1/2/3/4): 93-107.
- [11] SHAO W S, SHAO Z S, PI D C. Modeling and multi-neighborhood iterated greedy algorithm for distributed hybrid flow shop scheduling problem [J]. *Knowledge-Based Systems*, 2020, 194: 105527.
- [12] ZHENG J, WANG L, WANG J J. A cooperative coevolution algorithm for multi-objective fuzzy distributed hybrid flow shop [J]. *Knowledge-Based Systems*, 2020, 194: 105536.
- [13] MARICHEL VAM M K, PRABAHARAN T, YANG X S. Improved cuckoo search algorithm for hybrid flow shops scheduling problems to minimize makespan [J]. *Applied Soft Computing*, 2014, 19: 93-101.
- [14] MARICHEL VAM M K, TOSUN Ö, GEETHA M. Hybrid monkey search algorithm for flow shop scheduling problem under makespan and total flow time [J]. *Applied Soft Computing*, 2017, 55: 82-92.
- [15] LIU Y, YIN M H, GU W X. An effective differential evolution algorithm for permutation flow shop scheduling problem [J]. *Applied Mathematics and Computation*, 2014, 248: 143-159.
- [16] YING K C, LIN S W. Solving no-wait job-shop scheduling problems using a multi-start simulated annealing with bi-directional shift timetabling algorithm [J]. *Computers & Industrial Engineering*, 2020, 146: 106615.

- [17] 张洪亮,徐公杰,张金春.带有学习效应的多阶段混合流水车间调度问题研究[J].重庆师范大学学报(自然科学版),2021,38(1):87-95.
ZHANG H L,XU G J,ZHANG J C. Research on multi-stage hybrid flow shop scheduling problem with learning effect[J]. Journal of Chongqing Normal University (Natural Science),2021,38(1):87-95.
- [18] 秦旋,房子涵,张赵鑫.混合共生生物搜索算法求解置换流水车间调度问题[J].浙江大学学报(工学版),2020,54(4):712-721.
QIN X,FANG Z H,ZHANG Z X. Hybrid symbiotic organisms search algorithm for permutation flow shop scheduling problem [J]. Journal of Zhejiang University (Engineering Science),2020,54(4):712-721.
- [19] KOSZELEW J,KARBOWSKA-CHILINSKA J, OSTROWSKI K, et al. Beam search algorithm for anti-collision trajectory planning for many-to-many encounter situations with autonomous surface vehicles[J]. Sensors,2020,20(15):4115.
- [20] 顾勇,仲伟志,王磊,等.面向无人机毫米波高效波束搜索的码本优化设计[J].微波学报,2019,35(5):67-71.
GU Y,ZHONG W Z,WANG L, et al. Optimal code book design for efficient beam search of UAV millimeter wave communication[J]. Journal of Microwaves,2019,35(5):67-71.
- [21] BEAM J C. Genetic algorithm and random keys for sequencing and optimization[J]. ORSA Journal of Computing,1994,6(2):154-160.
- [22] TAILLARD E. Benchmarks for basic scheduling problems[J]. European Journal of Operational Research,1993,64(2):278-285.

Operations Research and Cybernetics

Permutation Flow Shop Scheduling Problem Based on Filtered Beam Search Algorithm

DONG Hai¹, WANG Hanpeng²

(1. School of Applied Technology, Shenyang University;

2. School of Mechanical Engineering, Shenyang University, Shenyang 110044, China)

Abstract: [Purposes] To solve the problem of minimizing the absolute deviation between the job completion time and the common due date in the permutation flow shop and minimizing the intermediate inventory. [Methods] A heuristic algorithm is proposed based on the filtered beam search method. First, build a mixed integer linear programming model based on minimizing absolute deviation and intermediate inventory, and use a two-stage method to solve the model; secondly, a Filtered beam search algorithm (FBSA) based on the elite selection strategy is proposed, which integrates the elite selection strategy and FBSA, and performs local and global searches on the iterative nodes with “hope” to determine the selected the optimal solution. [Findings] Finally, the FBSA based on the elite selection strategy is compared with other intelligent optimization algorithms in the three indicators of the optimal solution deviation percentage, average error rate, and improvement percentage through example simulation. The results show that the algorithm is efficient and stable in solving the proposed problem. [Conclusions] FBSA based on elite search strategy has good effects and advantages in dealing with PFSP.

Keywords: filtered beam search algorithm; permutation flow shop scheduling; mixed integer linear programming model; elite selection strategy; two-stage method

(责任编辑 方 兴)